Association for Information Systems
AIS Electronic Library (AISeL)

ACIS 2002 Proceedings

Australasian (ACIS)

December 2002

Events, Context, and Situations and Reactive Mobile Service Delivery

Paul O'Brien
University of Queensland

Jay Burmeister
University of Queensland

Events, Context, Situations and Reactive Mobile Service Delivery

Paul O'Brien

UQ Business School and School of Information Technology and Electrical Engineering
University of Queensland
p.f.obrien@mailbox.uq.edu.au

Jay Burmeister

School of Information Technology and Electrical Engineering
University of Queensland

Abstract

Mobile Users (MUs) require flexible, reactive service delivery due to their regularly changing location and activities and the lack of a wired Internet connection. A mobile service delivery system should be able to detect relevant events that occur such as change of location, availability of new last-minute specials, sales opportunities and safety issues and then reactively take action in response to these events. This paper describes a framework for delivering such a system. Issues addressed include MU and service states and events, context, situations and situation-action rules, and syntactically and semantically compatible XML schemas for their specification. A framework is proposed that is based on distributed, co-operating software agents and mobile data technologies.

Keywords

Software architecture, ubiquitous computing, situation management, software agents

INTRODUCTION

Mobile Users

As a person becomes more mobile, the need for and perceived value of services delivered to a mobile communication device increases. For example, people who rarely travel or who only travel short distances in their local area are unlikely to need sophisticated mobile travel reservation services. On the other hand, a mobile user may be regularly taking a number of flights, possibly international flights, on a single day, making a mobile travel reservation service more useful. As regularity of travel increases, the need for mobile communication services such as mobile email, and local information services such as traffic and weather alerts increases. As distances travelled increases the need for mobile transaction services such as travel reservation services and distribution services such as electronic maps and travel guides increases. This research is focussed on the delivery of reactive services, via mobile devices, to people who are highly mobile.

A mobile service delivery system that is able to detect user and service situations and to independently respond by invoking the required services on behalf of the user could provide value to the user while retaining maximum flexibility. Events that may create situations would include the user changing location, changes in exchange rates, availability of new last-minute specials, sales opportunities and safety issues.

Matching of service characteristics to user preferences is still a key requirement (O'Brien, 1999) of such a system. Adding the ability to determine the current location of the user at all times allows dynamic delivery of *mobile services* that are personalised on the basis of recorded preferences, location, context and relevant events as they occur.

Mobile Services

In the electronic commerce domain, the term *service* has many different meanings depending on the context in which it is used. It is most commonly used to refer to an *information service* that filters or searches for information on behalf of a user according to either explicitly stated or implicitly derived preferences. The term *web service* has also recently come into common usage to refer to the programmatic interfaces that are made for application-to-application communication (W3C, 2002).

The term *mobile services* used in this paper refers to the much broader set of services that are delivered by a service provider to its customers. This does include information services but it also includes communication, distribution and transaction services. Angehrn (1997) divided the virtual market space for an industry into these four quadrants to facilitate analysis. Angehrn's generic framework, the ICDT model, was developed to classify and illustrate significant business opportunities and threats generated by the Internet. The ICDT model splits the market space into four spaces, namely Information, Communication, Distribution and Transaction spaces. Angehrn treats each space separately because they each require different strategies and technological approaches.

Although Angehrn's model was developed to facilitate the analysis of the effects of the Internet on an industry from a marketing perspective, his model can also be useful to identify physical and virtual services that a particular industry typically provides. Similarly, Angehrn's model can be used to identify services in each space that are common for all mobile users regardless of industry. The ICDT model has been applied to generic mobile services to identify virtual services in each quadrant that are of value to mobile users.

Simply identifying the required services does not mean that they can be delivered in practice. Delivery of mobile services also requires lightweight, portable communication devices, widespread availability of mobile data telecommunications services and a common content encoding standard. Recent technology developments have now made this possible.

Recent Information Technology Developments

There have been dramatic advances recently in mobile data technologies that allow services to be delivered to mobile devices such as mobile phones and palm computers. WAP (Wap Forum, 2001) or I-Mode (NTTDocomo, 2002) based services are now available in Europe, the United States, Japan, Australia, most Asian countries, many South American countries and some African countries. These services have exploded onto the market over the last three years with varying success. Adoption in Europe, Japan and South-East Asia has been rapid. Growth of subscribers to the I-Mode and iAppli services in Japan has been consistently above 100,000 per week since it was introduced 3 years ago. There are currently over 30 million I-mode subscribers in Japan (NTTDocomo, 2002). The future potential of mobile data services has been demonstrated by the very successful I-mode service in Japan and WAP services in Northern Europe (Schaumann, 2000).

This success has been achieved despite the fact that the services are primarily delivered via mobile phones with very limited functionality, power and display capabilities. The availability of more sophisticated mobile communications devices and wearable computers through initiatives such as Symbian (Tivoli Systems Inc, 2001) and Bluetooth (Bluetooth SIG, 2001), will give users access to cost-effective devices and communications services for mobile data access wherever they are located. Furthermore, WAP version 2.0 (WAP Forum, 2001) has harmonised the I-mode and WAP content development and delivery standards. This is expected to significantly accelerate the adoption of mobile data services outside Japan.

Focus Of This Research

The literature review and exploratory interviews with a small group of MUs indicate that some services are perceived to be of high value if they can be delivered to MUs during their trip. The design of a framework that provides the ability to deliver information that is dynamically filtered, make recommendations and take independent action as the MU's current context changes is the focus of this research.

The framework is based on context-aware, situation-driven, co-operating software agents and mobile data technologies. As a proof of concept a prototype of a reactive travel service is being built that will test the performance of the design in each of the four virtual market spaces for travel. This reactive travel service will be delivered through mobile data appliances thereby providing a reactive mobile travel service

REACTIVE MOBILE SERVICE DELIVERY

The key to the provision of an effective, high added value mobile service is the ability to detect *relevant* events and to independently take the action that is most appropriate for the resulting *situation*. Detection of an event requires an awareness of previous and subsequent states of MUs and services. Determination of the most appropriate action to take after an event or series of events requires the ability to detect the occurrence of *relevant situations* that is relevant states of affairs existing within a relevant context. This section discusses the theory of states, events, contexts, situations and situation-response rules. Some simple examples of MU situations and appropriate situation-response rules are then provided.

States

The state of an entity can be defined as the set of current values of all the properties of the entity. When we represent an entity in an information system we choose the subset of properties of the entity that are relevant to the focus of the system and represent those properties using agreed attributes. The state of the representation of the entity would therefore be the current set of values of the relevant attributes of the entity, its state variables. The allowed or lawful states of an entity are limited by its current context. For example, “sleeping” is an unlawful state for a person whose current activity is “skiing”. Entities may also be related to other entities within the domain of interest such that a change of state of an entity (or event) may cause a change of state of a related entity (Weber, 1997). In the proposed framework, the state of each user, service and context must be monitored for relevant changes, or *events*.

Events

An event (or state transformation) occurs when an entity changes its state. That is, at least one of its properties changes. A state transformation of one entity may also cause a state transformation in a related entity. The related entities may be within the same system (internal agents) or external to the system but coupled to an entity within the system (external agents). The external agents make up the context of the system (Weber, 1997). For example, a MU changes her state when she travels from one location to another. This event may then trigger a change of *context* that in turn causes a safety alert from a safety monitoring system (internal agent). Similarly, a change in the availability of accommodation (external event) in a particular location may cause the MU to move to another location, generating a MU event (internal).

In a reactive mobile service delivery system the current state and current context of the MUs must be monitored constantly so that significant events in either can be propagated to related internal and external agents. Commercial travel intelligence services such as WorldCue (iJet, 2000) provide comprehensive information and alerts about external events to travel agents, and in future directly to MUs. However, they do not propagate MU events to external agents or travel agents. In this situation, when an unexpected change in the MU’s state occurs, the onus is on the MU to explicitly advise his agent or the external service provider. This assumes that the MU is aware of the consequences of the state change and is capable of advising his external service provider and/ or agent. Sickness, accident and lack of communication services may prevent this. It is therefore essential that the MU’s virtual consultant and/ or external service providers be *automatically* advised of any *relevant* change in a MU’s state.

A MU’s state is determined by the current values of her properties or state variables. Relevant MU properties would include a unique identifier such as a URL, current location, current activity and preferences (Kanter, 2000). A change in the value of any of these state variables will generate a *MU event*.

The state of a service is also determined by the current values of its properties or state variables. Relevant service properties would include a unique identifier such as a URL, location, availability, price and season. A change in the value of any of these state variables will generate a *service event.*

In addition, a clock tick event will occur each second. This may result in a change of context, for example to a new day or season, which would generally result in one or more changes in context.

Context

Decisions and assertions that are made regarding a MU or service must consider the current state and context. For example, information about a MU's current location and activity only allows decisions and assertions to be made about this location at this time (Schmidt *et al*., 1999). It does not allow decisions to be made about other times and locations (Dey and Abowd, 2000; Dey *et al*., 1998).

Lenat (1998) argues that context has twelve basic dimensions, rather than the two dimensions commonly used by researchers, absolute time and absolute place. For example, Lenat includes a dimension, Type of Place that is used to specify a non-absolute type of place such as "in bed".

Assertions about an entity are generally only applicable in specific contexts. Therefore, an answer to a question about an entity must be a pair <context, answer>. That is, "the answer is…in this context". For example, two possible answers to the question "How much is a single standard room in the Hyatt Coolum" would be <low season, $150> and <high season, $250> (Lenat, 1998).

Lenat's (1998) experience with contexts or micro-theories has shown that it results in simpler assertions and extensive re-use of assertions but also imposes the additional burden of choosing the most appropriate context from a large number of small contexts. Consequently, a trade-off needs to be made between coding time within the contexts and time to select the best context. Mobile data technologies allow data, such as location, to be captured that simplifies the selection of the most appropriate context and hence, simplifies the automated decision-making process.

By considering the MU's current context, existing alerting services, such as WorldCue (iJet, 2001), could improve the relevance, quality and accuracy of their alerts. However, the ability to eliminate irrelevant, unnecessary and inappropriate messages and alerts automatically is not only dependent on context awareness. It also requires *situation awareness*.

Situations

Etzion and Adi (2000) define a situation as "a reactive entity that receives events as an input, combines composition filtering with content filtering, and detects situations as an output". When any event occurs, changes occur in the state variables of affected entities. The set of new states and the context in which they occur creates a new situation. If the new situation is relevant to the domain of interest then a rule needs to be defined that specifies the appropriate action that should be taken in response to occurrences of the situation (Adi *et al*., 2000). Where the same response is required for a number of situations, these situations form a class or type of situation (Cherry, 2001). For each relevant *type or class* of situations a rule must be defined to specify the required response (Cherry, 2001). Sequences of situations make up a scenario.

Detection of situations requires awareness of the context dimensions of Absolute Time, Type of Time, Absolute place, Type of place, Culture, Granularity, and Local bindings of variables. Etzion and Adi (2000) also propose the use of a relative timespan measure, lifespan, rather than absolute time, to define a time interval between two events during which situation detection is relevant. For example the timespan for a *high season* situation would be from the clock tick event at midnight on the last day of the previous season to the clock tick before midnight on the last day of the high season. Applying Lenat's granularity dimension to absolute time can accommodate this.

Cherry (2001) proposes that objects, applications, subsystems and independent functions are "machines whose rules map situations (facts about their representations) onto appropriate actions". He calls these machines Situation-Action Machines. For each relevant situation, a Situation-Action rule of the form "if situation then action" must be defined. That is, when a particular situation occurs, actions are taken that are appropriate to handle that type of situation. Table 3 lists a number of (but not all) situations that may occur for transport services and the corresponding response rules for each.

The Situation-Action machine is defined by identifying relevant *types* of situations and defining action rules for each situation type. The actions taken may, in turn, create new situations that also require actions to be taken. In a complex system it would not be feasible to identify at design time, every possible *combination* of situations and events that could occur. However, the principles of game theory (Binmore, 1992) indicate that a suitable overall response can be expected by identifying the possible situation types, and specifying the best possible response to each individual situation type *in isolation*.

Relationship Between Events, Context and Situations

Figure 1 summarises the relationship between events, states, contexts, situations and actions.

$e \rightarrow \delta s$
$\delta s \Rightarrow \delta C$ OR $\delta s \Rightarrow S$
$\delta C \Rightarrow S$
$S \rightarrow A1, A2 \ldots An$

where e = event, δC = change in context, δs = change in state, S = situation, A = action, $\rightarrow$ = causes and $\Rightarrow$ = *may* cause

Figure 1: Relationship between events, context, situations and actions

Events cause changes in the state variables of a user or a service. The change in state may cause a change in context, which in turn may cause a situation to occur. The change in state my also directly cause a situation to occur *without* a change in context. In response to the occurrence of a situation one or more actions are taken.

Events can be of 3 types, user events, service events or temporal events. Temporal events occur independently. At the smallest level of time granularity of interest, the only temporal events are clock ticks. These clock ticks cause a change of state of time. Every 60th tick the minute changes, which may cause the hour, day, month, year, season etc. to change.

Event	Type	Source	New Context	Situation	Possible Action (s)
Clock tick. 23:59:59 to 24:00:00	Temporal	External	Time: 20 July, Type of time: non-workday	N/A	
Clock tick. 24:00:00 to 24:00:01	Temporal	External	Time: 21 July, Type of time: Sabbath	Working on Sabbath	• Send reminder to stop working
			Time: 21 July, Type of time: Low Season	New accommodation available	• If best price in range, change reservation
User arrives in Brisbane	Spatial	User	Location: Brisbane	Customer arrival	• Send discount limo offer
User changes preferred accommodation price range	User State Change	User	No change	Suitable accommodation available	• Send accommodation offer
Flight departure time changed from 9:00 to 11:00	Service State Change	Transport Service	No change	Same day flight delay	• Alert MU of new departure time; • Reschedule transfers
Flight departure time changed from 19:00 22/7 to 9:00 23/7	Service State Change	Transport Service	Time:23 July	Next day flight delay	• Alert MU of new departure time; • Reschedule transfers • Reserve accommodation

Event	Type	Source	New Context	Situation	Possible Action (s)
					for tonight
Flight cancelled	Service State Change	Transport Service	No change	Flight cancellation	• Reserve seat on another flight; • Alert MU of new flight details; • Reschedule transfers; • Reserve accommodation for extra night(s)
New flight reservation	User state change	User	No change	Confirmed reservation	• Alert MU of flight details; • Arrange transfers;
Transfer arrangements changed	Service state change	Transport Service	No change	Transfer arrangements altered	• Alert MU of transfer details;
Conflict broken out in Solomon Is	Service state change	Safety Service	No change	Danger in location on future itinerary	• Send safety alert to MU; • Cancel trip to Solomon Is

Table 1: Examples of Relationship between Events, Context, Situations and Actions

Some examples are given in Table 1. At the various levels of temporal granularity there may be resultant context changes. For example, if the new time is 1 second after midnight then the season may change, the end of a work shift may occur (context change from working to leisure context), or the new day may be a different *type* of day such as weekend or Sabbath.

Each of these new contexts may create situations that require actions. For example, change of season may require changes to travel or accommodation bookings. These actions may then generate additional, non-temporal events and so on. Each external event may introduce additional instability to the system by creating situations to which there must be a response.

To respond to situations independently, a system must be operating in a *situation management framework* that allows it to detect service, user and temporal events, context changes and any resulting situations and then independently take the appropriate actions according to a set of situation-response rules.

SITUATION MANAGEMENT FRAMEWORK

In the proposed situation management framework, a change in state of a relevant *service* is detected by the associated Service Provider Agent (SPA) and then propagated to the Service Management Agent (SMA) for that *type* of service. This will create a new situation within the SMA. If the situation is relevant to any of the registered User Agents (MUAs), then the SMA is required to take action on behalf of the MUA by applying the appropriate response rules. For most situations the response will simply be to pass a message through to the MUA notifying it of the situation. However, some situations that would normally be handled by a physical agent may require actions to be taken for the MU by the SMA. The specification of the relevant situations and response rules form the core of the reactive mobile service delivery system.

This section describes the situation management framework and its components, Mobile User Agents, Service Agents and Service Management Agents. It also discusses the required specification languages and gives examples of how they are used.

Co-operating Multi-agent Architecture

The situation management framework (Figure 2) is comprised of three types of software agents, a *service provider agent* (SPA) for each particular service, a *service management* *agent* (SMA) that monitors all the SPAs for a particular *type* of service and a *mobile user agent* (MUA) that provides a persistent online presence for each registered MU. Each of the three types of software agents independently performs specialised functions in co-operation with the other types of agents.

Service Provider Agent

- Monitors a *particular* online or external service such as wcities (Wcities Corp., 2001) for significant changes

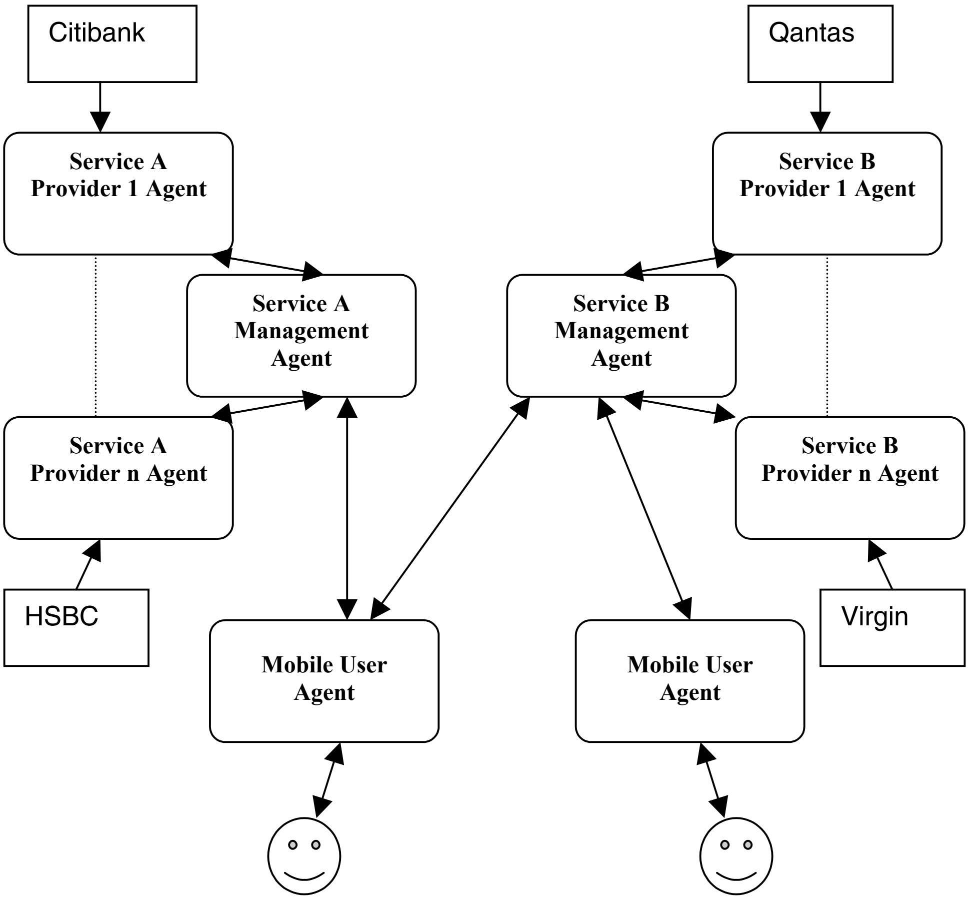

Figure 2: Mobile Service Delivery Framework

- Converts information regarding changes in a *particular* service into the agreed interchange format
- Sends the required event information to the Service Management Agent for that *type* of service

Service Management Agent

- Requests/ obtains/ parses/ monitors a particular *type* of information from service provider agents using agreed message formats
- Can request information from MUAs and SPAs
- Filters information and alerts coming from SPAs and MUAs and takes appropriate actions such as generating a filtered alert or changing a reservation
- Monitors absolute time, MU events and service events and changes the current context if required

Mobile User Agent

- Collects MU data from the MU's communication device and sends alerts generated by MU events to SMAs
- Collects filtered data and alerts generated by SMAs, renders it into the appropriate format and sends it to the MU's communication device when it is online

Specification Languages

Etzion and Adi (2000), Cherry (2001) and Lenat (1998) use specification languages that are syntactically and semantically incompatible. An important output of this research is the definition of XML schemas that are syntactically and semantically compatible and are based on Etzion and Adi, Cherry and Lenat's work. These schemas will allow the unambiguous definition of contexts (Context Definition Language), states and events (Event Specification Language), situations and responses to detected situations (Situation Definition Language).

Examples of the use of each of these languages to encode contexts, states, events, situations and situation-response rules are given in the Appendix.

CONCLUSION

This paper introduced the concept of a mobile user and proposed a framework for the ubiquitous delivery of reactive services to mobile users. The framework brings together previous work on states, events, contexts, situations and situation-response rules. In doing so, it was identified that there was a need for a set of syntactically and semantically compatible languages to define each of these. This is one of the contributions of this work. The examples illustrate the simplicity with which these concepts can be integrated.

LIMITATIONS

The mobile communications infrastructure in most locations is very limited with the exception of major cities in Europe, the United States, Japan, Singapore, Hong Kong and Korea. Similarly, the large size and weight, small screen size, limited processing power and high cost of current mobile devices limits their attractiveness to budget and weight conscious MUs. This will limit the effective deployment of the system in the short term but will not prevent the development and testing of a prototype.

It is expected that advances in voice recognition and speech synthesis technology will soon make its deployment to mobile devices more practical. This will significantly enhance their usability.

OPPORTUNITIES FOR FURTHER RESEARCH

Although the proof of concept prototype will only include four service types, the system framework will allow it to be extended to include other service types as well as to any activities undertaken by mobile people.

REFERENCES

Adi, A., Botzer, D. & Etzion, O. (2000): Semantic Event Model and Its Implication on Situation Detection, *Proceedings of the 2000 European Conference on Information Systems.*

Angehrn, A. (1997): The Strategic Implications of the Internet, *Proceedings of the 1997 European Conference on Information Systems.*

Binmore, K. (1992): *Fun and Games*: *A Text on Game Theory,* Lexington, MA: Heath

Bluetooth SIG (2001) at http://www.bluetooth.com

Cherry, G. (2001) *Extending and Formalizing The Object-Oriented Paradigm with Situation-Driven Modeling*, Thought**Tools Corp, Cambridge, Massachusetts at www.sdml.com

Dey, A. & Abowd, G. (2000): CybreMinder: A Context-Aware System for Supporting Reminders, *Proceedings of the Second International Conference on Handheld and Ubiquitous Computing (HUC 2000)*, pp172-186, Springer-Verlag, Heidelberg.

Dey, A., Abowd, G. & Wood, A. (1998): *CyberDesk: a framework for providing self-integrating context-aware services,* Knowledge-Based Systems, 11, pp3-13, Elsevier, Amsterdam.

Etzion, O.. & Adi, A. (2000) *The situation manager - an application development tool for reactive applications*, Research White Paper, IBM Research Laboratory, Haifa, at http://www.haifa.il.ibm.com/projects/software/publications.html

IJet Travel Intelligence Inc. (iJet) (2001): *WorldCue*, at http://www.ijet.com

Kanter, T. (2000) Event-Driven, Personalizable, Mobile Interactive Spaces, *Proceedings of the Second International Conference on Handheld and Ubiquitous Computing (HUC 2000),* pp1-11, Springer-Verlag, Heidelberg.

Lenat, D. (1998): *The Dimensions of Context-Space,* Cycorp, Austin, at *www.cyc.com.*

NTTDocomo (2001): *The I-Mode Service*, at http://www.nttdocomo.com.

O'Brien, P. (1999): *Intelligent Assistants For Retail Travel Agents*, International Journal of Information Technology and Tourism, Vol. 6 No. 3/4, Cognizent, New York

Schaumann, J. (2000): *WAP vs. I-MODE*, at http://www.netmeister.org/palm/WAP_iMODE

Schmidt, A., Beigl, M. & Gellersen, H. (1999) *There is more to context than location,* Computers & Graphics 23, pp 893 – 901, Pergamon

Tivoli Systems Inc. (2001): *IBM and Symbian to Deliver the First Wireless Solution to Extend Today's e-business Infrastructures to Next Generation Smartphones*, at http://www.tivoli.com/news/press/pressreleases/en/2001/0322_nokia.html

W3C (2002): *The Web Service Activity*, at http://www.w3.org/2002/ws

WAP Forum (2001): At http://www.wapforum.org

Weber, R. (1997): *Ontological Foundations of Information Systems*, Coopers & Lybrand and the Accounting Association of Australia and New Zealand, Melbourne.

Wcities Corp. (2001): *Wcities geo-coded content service*, at http://www.wcities.com

APPENDIX

Context Definition Language

```
<context>
    <context-id> c1 </context-id>
    <absolute-time>1450 09042002</absolute-time>
    <type-of-time> work-time </type-of-time>
    <absolute-place> 1321234324345,2312342344</absolute-place>
    <type-of-place> office </type-of-place>
  <granularity>
            <time-granularity>day</time-granularity>
            <place-granularity>city</place-granularity>
        </granularity>
        <local-bindings>
            <local-name>Australia</local-name>
            <local-language>English</local-language>
            <local-religion>Christian</local-religion>
            <local-ethnicity>multicultural</local-ethnicity>
            <local-currency>AUD</local-currency>
            <local-average-income>40,000</local-average-income>
        </local-bindings>
```

```
</context>
```

MU State Definition Language

```
<MU-state>
        <MU-ID>h1</MU-ID>
        <MU-name>Paul O'Brien</MU-name>
        <MU-language>English</MU-language>
        <MU-language>French</MU-language>
        <MU-religion>Catholic</MU-religion>
        <MU-ethnicity>Anglo-Saxon</MU-ethnicity>
        <MU-income>
                <MU-currency>AUD</MU-currency>
                <MU-amount>70,000</MU-amount>
        </MU-income>
        <MU-preferences>
                <MU-airline>Qantas</MU-airline>
                <MU-hotel-rating>4*</MU-hotel-rating>
                <MU-hotel-chain>Accor</MU-hotel-chain>
                <MU-travel-time>evening</MU-travel-time>
        </MU-preferences>
        <MU-reservation>
                <carrier>Qantas</carrier>
                <flight-id>Q1</flight-id>
                <departure-place>bne</departure-place>
                <arrival-place>syd</arrival-place>
                <departure-date>09042002</departure-date>
                <departure-time>2250</departure-time>
                <arrival-date>09042002</arrival-date>
                <arrival-time>2250</arrival-time>
        </MU-reservation>
</MU-state>
```

Event Definition Language

```
<event>
        <event-id> e1 </event-id>
        <event-type> MU-Reservation </event-type>
        <event-start-state>
                <MU-ID>h1</MU-ID>
        </event-start-state>
        <event-end-state>
                <schedule>
                        <carrier>Qantas</carrier>
                        <flight-id>Q1</flight-id>
                        <departure-place>bne</departure-place>
                        <arrival-place>syd</arrival-place>
                        <departure-date>09042002</departure-date>
                        <departure-time>2250</departure-time>
```

```
            </schedule>
        </event-end-state>
</event>
<event>
        <event-id> e2 </event-id>
        <event-type> same-day-flight-delay </event-type>
        <event-start-state>
                <schedule>
                        <carrier>Qantas</carrier>
                        <flight-id>Q1</flight-id>
                        <departure-place>bne</departure-place>
                        <arrival-place>syd</arrival-place>
                        <departure-date>09042002</departure-date>
                        <departure-time>2250</departure-time>
                </schedule>
        </event-start-state>
        <event-end-state>
                <schedule>
                        <departure-time>2350</departure-time>
                </schedule>
        </event-end-state>
</event>
<compound-event>
        <compound-event-id>ce1</compound-event-id>
        <event-id> e1 </event-id>
        <relation> and </relation>
        <event-id> e2 </event-id>
</compound-event>
```

Situation Definition Language

```
<situation-type>
        <situation-type-id> s1 </situation-id>
        <situation-type>transfer-required</situation-type>
        <context-id> c1 </context-id>
        < has-reservation>yes<has-reservation>
                            <event-type> same-day-flight-delay </event-type>
        <action>
                <action-id>a1</action-id>
                <action-name>MU-departure-time-alert</action-name>
                <action-description>Send departure time alert to MU</action-description>
                <action-URL>www.uq.edu.au/MUActions/departure-time-alert.asp</action-URL>
        </action
        <action>
                <action-id>a2</action-id>
                <action-name>reschedule-transfer</action-name>
```

<action-description>Reschedule transfer to transport departure point</action-description>

<action-URL>www.uq.edu.au/MUActions/arrange-transfer.asp</action-URL>

</action

<situation-type>

COPYRIGHT